

**C4ISR Architectures, Social
Network Analysis and the FINC
Methodology: An Experiment in
Military Organisational Structure**

Anthony H. Dekker

DSTO-GD-0313

DISTRIBUTION STATEMENT A:
Approved for Public Release -
Distribution Unlimited

20020621 043

C4ISR Architectures, Social Network Analysis and the FINC Methodology: An Experiment in Military Organisational Structure

Anthony H. Dekker

**Information Technology Division
Electronics and Surveillance Research Laboratory**

DSTO-GD-0313

ABSTRACT

In this paper we present a simple testbed for experimenting with C4ISR architectures (based on a "SCUD hunt" scenario), the FINC methodology for analysing C4ISR architectures, and some experimental results. The testbed allows us to explore different organisational architectures under a range of conditions. The FINC (Force, Intelligence, Networking and C2) methodology allows the calculation of three metrics for every C4ISR architecture. Applying the FINC methodology to our testbed provides a partial validation of the methodology, as well as allowing us to derive four basic principles of C4ISR architectures.

RELEASE LIMITATION

Approved for public release

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION

DSTO

AQ F02-09-1843

Published by

*DSTO Electronics and Surveillance Research Laboratory
PO Box 1500
Edinburgh South Australia 5111 Australia*

*Telephone: (08) 8259 5555
Fax: (08) 8259 6567
© Commonwealth of Australia 2002
AR-012-129
January 2002*

APPROVED FOR PUBLIC RELEASE

C4ISR Architectures, Social Network Analysis and the FINC Methodology: An Experiment in Military Organisational Structure

Executive Summary

In responding to the Revolution in Military Affairs and rapid change in the modern strategic environment, it is important to utilise the best possible C4ISR architectures for the Australian Defence Force. Consequently, it is extremely important to evaluate the effectiveness of different C4ISR architectures. This can be done using the regular series of military exercises. However, these are not capable of examining the impact of technologies not yet in service. Wargaming is capable of examining such technologies, but both wargaming and real exercises have a substantial cost, and therefore there is considerable benefit in a low-cost methodology for evaluating C4ISR architectures, and selecting for further experimentation those which the methodology identifies as the best candidates. The FINC (Force, Intelligence, Networking and C2) methodology satisfies this goal.

The FINC methodology allows the calculation of three metrics for every C4ISR architecture: the **information flow coefficient** measuring tempo superiority, the **coordination coefficient** measuring coordination superiority, and the **intelligence coefficient** measuring information superiority.

Like all methodologies, the FINC methodology requires validation, and this report describes the first step in validating it. For this first step, we utilise a testbed (based on a "SCUD hunt" scenario) which is simple, yet allows us to explore the impact of different organisational architectures under a range of different conditions.

Applying the FINC methodology to our experimental testbed allows us to derive four basic principles of C4ISR architectures, and an indication of which military organisational structures are appropriate for different tempo/information quality regimes. Our experiments indicate that at slow to moderate tempo with poor sensors, intelligence superiority (indicated by a high intelligence coefficient) is the most critical.

At slow tempo with fair to good sensors, coordination superiority (indicated by a low coordination coefficient) is also important, and a highly centralised architecture (such as the use of highly centralised Air Tasking Orders by the US Air Force) performs well in this regime.

At moderate tempo with fair to good sensors, tempo superiority (indicated by a low information coefficient) is also important, and network-centric warfare seems to perform well in this regime.

At high tempo, coordination superiority is less important than intelligence superiority and tempo superiority, and taking time to achieve perfect coordination may be detrimental in this regime. Table (i) summarises these results.

Table (i): Summary of Results

	Poor sensors	Fair to Good sensors
Slow tempo	Region A: <i>Information superiority</i> is most important (high intelligence coefficient) (high intelligence coefficient)	Region B: Balance <i>information superiority</i> and <i>coordination superiority</i> (high intelligence coefficient and low coordination coefficient)
Moderate tempo		Region C: Balance <i>all three kinds</i> of superiority (high intelligence coefficient, low information flow coefficient, and low coordination coefficient)
Fast tempo	Region D: Balance <i>information superiority</i> and <i>tempo superiority</i> (high intelligence coefficient and low information flow coefficient)	

These results were obtained by analysing performance of different C4ISR architectures for our experimental testbed. Figure (i) shows two of the eight architectures examined. All eight architectures are defined in detail in the body of the report.

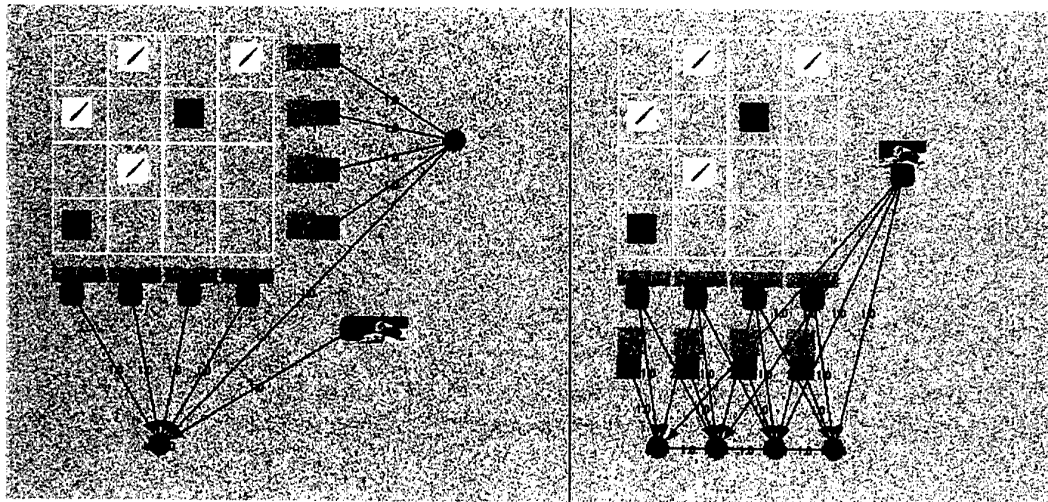


Figure (i): Two Alternatives Architectures for Experimental Testbed

Contents

1. INTRODUCTION	1
1.1 C4ISR Architectures.....	1
1.2 Social Network Analysis.....	2
2. THE EXPERIMENTAL TESTBED	3
2.1 The Centralised Architecture Without Information Sharing	4
2.2 The Split Architecture Without Information Sharing	5
2.3 The Distributed Architecture Without Information Sharing	6
2.4 The Negotiation Architecture Without Information Sharing.....	7
2.5 Architectures with Information Sharing.....	8
3. EXPERIMENTAL RESULTS.....	11
4. THE FINC (FORCE, INTELLIGENCE, NETWORKING AND C2) METHODOLOGY.....	12
4.1 Delay Analysis 1: the information flow coefficient	14
4.2 Delay Analysis 2: the coordination coefficient	15
4.3 Centrality Analysis	15
4.4 Intelligence Analysis: the intelligence coefficient	16
5. MODELLING THE TESTBED WITH THE FINC METHODOLOGY	18
5.1 Explaining the Equation of Best Fit (a).....	20
5.2 Explaining the Equation of Best Fit (b)	20
5.3 Explaining the Equation of Best Fit (c).....	21
5.4 Explaining the Equation of Best Fit (d)	21
6. FOUR BASIC PRINCIPLES OF C4ISR ARCHITECTURES	21
6.1 Region A: Slow to Moderate Tempo and Poor Sensors — Intelligence Superiority Critical	22
6.2 Region B: Slow Tempo and Fair to Good Sensors — Intelligence & Coordination Superiority.....	22
6.3 Region C: Moderate Tempo and Fair to Good Sensors — Combined Superiority.....	22
6.4 Region D: Fast Tempo — Intelligence & Tempo Superiority	23
7. CONCLUSIONS	24
8. ACKNOWLEDGEMENTS	24
9. REFERENCES.....	24
APPENDIX A: TABLES 8 AND 9.....	27

1. Introduction

In this paper we discuss the application of Social Network Analysis concepts [1, 2] to C4ISR (Command, Control, Communications, Computers and Intelligence, Surveillance & Reconnaissance) architectures. In particular, we describe the FINC methodology [3], which calculates a number of metrics for evaluating C4ISR architectures.

We also present a simple experimental testbed for evaluating some C4ISR architectures, and some experimental results which validate the FINC methodology. We believe that, in order to gain understanding of organisational design, it is important to use simple and easily understood testbeds that permit rapid experimentation, in much the same way that early experiments with rapidly-breeding fruit flies [4] led to modern successes in genetic engineering. We share this belief with the Carnegie Mellon group, which has published extensive studies using a simple testbed focused on bureaucratic organisations [5, 6]. The use of agent-based distillations for studying military operations [17], and Conway's famous "Game of Life" are based on similar philosophies of experimentation.

For our purposes, we require a testbed more closely aligned with military activities, and which models an organisation not only as an information-processing structure, but which also includes the interactions between the organisation and its environment. The organisation receives information (intelligence) from its environment, makes decisions, and produces some effect on the environment. In this way, it resembles a model of a biological organism, and like an organism, the performance of a military organisation depends on the appropriateness of its response to its environment. The testbed we use is based on the SCUDHunt [7] game, which was originally designed for experiments on the coordination of virtual (distributed) teams using different communication tools. This testbed is precisely suited to our purpose since it is simple, yet has an interesting range of behaviour, allowing us to explore the impact of different organisational architectures under a range of different conditions.

This paper is divided into three main parts: the first discusses our experimental testbed and the results of experimentation, the second describes the FINC methodology, and the third applies the FINC methodology to our experimental results, allowing us to derive four basic principles of C4ISR architectures, which are outlined in table 7 at the end of the paper.

1.1 C4ISR Architectures

The term C4ISR architecture [8] is used by the US and other militaries to refer to the organisational structure used by military forces in carrying out a mission. The key aspect of C4ISR is **command** (authority and responsibility) and **control** (exercising authority over subordinates). These two indivisible aspects of leadership are referred to as C2. Since **communications** and **computer** technology are important in carrying out these leadership functions in a large organisation, the acronyms C3 and C4 are used to include these facilities.

Since leadership cannot be carried out without information of some kind, the acronyms C3I and C4I are used to include **intelligence**, which means the collection of relevant information. The acronym C4ISR includes two specific sources of information: **surveillance** (systematic observations of something) and **reconnaissance** (observations on a specific occasion).

Traditionally military structures have been very hierarchical, but modern innovations in communications and computer technology have made a wide range of other structures possible. At the same time, an emerging emphasis on operations other than war may require more flexible non-traditional organisational structures. In this environment, there is a need for formal techniques for the evaluation of a wide range of organisational structure options. We believe that Social Network Analysis techniques are the obvious choice for such evaluation.

1.2 Social Network Analysis

Social Network Analysis is an approach to analysing organisations focusing on the relationships between people and/or groups as the most important aspect. Going back to the 1950's, it is characterised by adopting mathematical techniques especially from graph theory [9, 10]. It has applications in organisational psychology, sociology and anthropology. A good summary is found in [1].

The first goal of Social Network Analysis is to visualise communication and other relationships between people and/or groups by means of diagrams. The second goal is to study the factors which influence relationships and to study the correlations between relationships. The third goal is to draw out implications of the relational data, including bottlenecks where multiple information flows funnel through one person or section (slowing down work processes) and situations where information flows does not match formal group structure. The fourth and most important goal of Social Network Analysis is to make recommendations to improve communication and workflow in an organisation, and (in military terms) to speed up the orient-observe-decide-act (OODA) loop or decision cycle.

Social Network Analysis provides an avenue for analysing and comparing formal and informal information flows in an organisation, as well as comparing information flows with officially defined work processes. In previous work, we have applied Social Network Analysis to military organisations in more or less standard ways [2]. In this paper, we describe an extension to traditional Social Network Analysis for the specific area of C4ISR architectures which we call the FINC (Force, Intelligence, Networking and C2) methodology. We have constructed a Java-based tool called CAVALIER [18], to carry out this and other forms of Social Network Analysis.

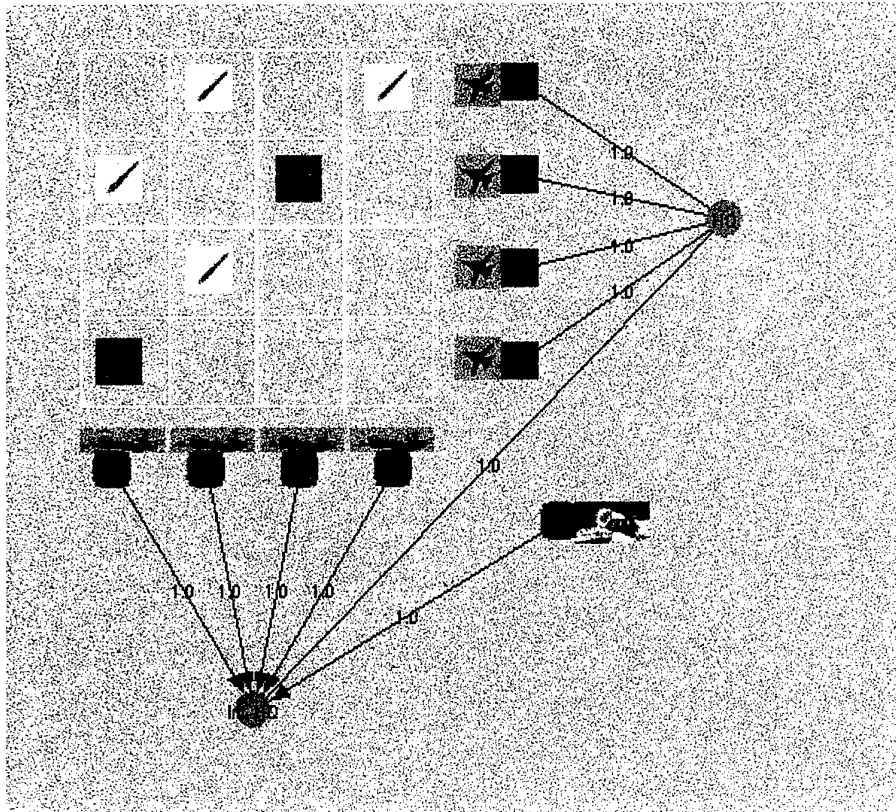


Figure 1: Experimental Testbed: Centralised Architecture

2. The Experimental Testbed

The testbed we use is based on the SCUD Hunt [7] game, which was originally designed for experiments on the coordination of virtual teams. The area of operations is a 4x4 grid which contains four randomly-located missile launch sites (see figure 1). Two kinds of sensors are used to locate the launch sites: four surveillance aircraft fly along the columns of the grid, producing intelligence on the columns, while a satellite provides intelligence on the entire grid. Both kinds of sensors have a quality factor q (ranging from 0.1 to 0.9), and on each grid square sensors fail with probability $1-q$, reporting a missile launch site even if none exists in that square. Consequently, the intelligence data may include "ghost" missile launch sites ("false alarms"), as shown in figure 1 (the figure is produced by our Java-based CAVALIER tool). There are no failures to detect targets, i.e. every target is detected, so that there will always be at least four targets reported. Reported positions for the real targets are also completely accurate (i.e. there are no ambiguities in location). Combining intelligence sources (sensor fusion) gives a more accurate picture than single sensors would, since the chance that two sensors will fail simultaneously on the same grid square is reduced. The testbed is programmed to ensure that two launch sites never occupy the same grid square, so that the sensor fusion process is trivial.

There are four fighter squadrons that must destroy the missile launch sites. Each fighter squadron can be assigned to only one (real or ghost) launch site, so that the

presence of ghost launch sites reduces the chance that the real sites will be destroyed. To complicate matters further, the air strikes occur after a delay due to planning. Planning requires a number of **time steps**, and at each time step, there is a probability p (ranging from 0.01 to 0.5) that any particular launch site will move to a safe location, in which case the fighter strike will be unsuccessful. The probability p thus provides an indirect measure of battlefield **tempo**: at a low value of $p = 0.01$ (slow tempo), there is a better than even chance that launch sites will remain at the location where they were detected, even after 68 time steps. Hence rapid response is not critical in this case. At moderate values of $p = 0.05$ or $p = 0.1$ (moderate tempo), there is still a better than even chance that launch sites will remain at the location where they were detected after 6 time steps. However, at the high value of $p = 0.5$ (fast tempo), after only one time step there is an even chance that launch sites will have moved to a safe location, after two time steps this has become a 75% chance, after three time steps, 87.5%, etc., making rapid response extremely critical in this case.

For each run of the experiment, we choose four randomly-located launch sites, and simulate a single strike mission. The measure of performance is the number of missile launch sites destroyed (which will range from 0 to 4). To produce statistically valid results, for each C4ISR architecture and combination of p and q , we average the performance of 10,000 experimental runs.

This experimental testbed satisfies our “fruit fly” condition of being simple and easily understood, and enables us to explore the impact of different C4ISR architectures under all possible combinations of p and q . For this experiment, we study the performance of eight C4ISR architectures: **centralised**, **split**, **distributed**, and **negotiation**, with and without **information sharing**.

2.1 The Centralised Architecture Without Information Sharing

In the centralised architecture (see figure 1), intelligence data (from the surveillance aircraft and the satellite) is collected by an intelligence headquarters (Int HQ) and combined to produce a reference intelligence picture, and a list of targets (which includes the four real missile launch sites plus zero or more ghost sites). The list of targets is passed to the top-level headquarters (HQ) which chooses four of the targets and assigns them to the four fighter squadrons (in US Air Force doctrine, this is called an Air Tasking Order, or ATO). Air strikes occur 5 time steps after intelligence collection begins.

The centralised architecture is the preferred model of the US Air Force, which generally operates with air superiority in support of joint operations (see [11] for a description of the air campaign during the Gulf War). The US Air Force possesses very good communications and intelligence capability (particularly AWACS aircraft) and the inherent speed of aircraft makes it relatively easy to position assets in the battle space. Air superiority allows the US Air Force to set the tempo of the battle. All these factors make the centralised architecture a good choice for the US Air Force, as we will see later.

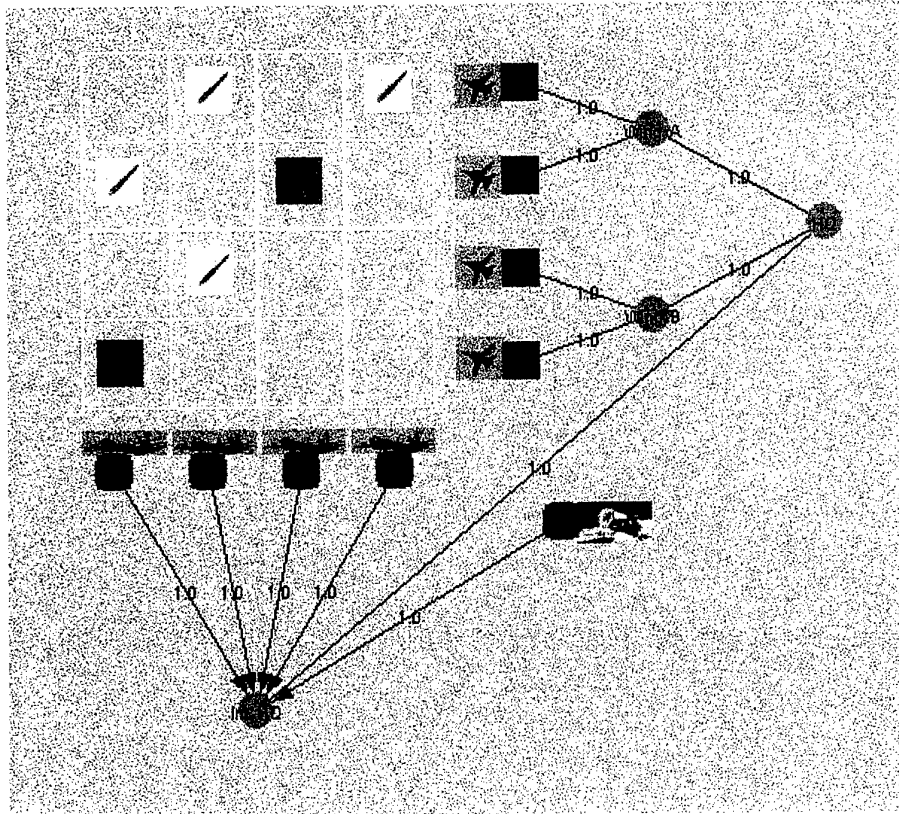


Figure 2: Split Architecture

2.2 The Split Architecture Without Information Sharing

The split architecture (see figure 2), is similar to the centralised architecture, but the top-level headquarters only partitions the battlespace between two **wings** (each consisting of two squadrons). This is done by drawing a vertical line through the grid so that there are roughly the same number of targets on either side (usually, but not always, this vertical line will pass through the centre of the grid). The left and right sides of the partition are assigned to the two wings, and each wing headquarters then assigns its two squadrons to two of the targets in its area of responsibility. Because of the additional level of command, with this architecture air strikes occur 6 time steps after intelligence collection begins.

The split architecture is essentially the traditional land force structure. Land operations usually involve problems which are too complex for centralised optimisation, and so benefit from being hierarchically subdivided [16]. The split architecture thus provides some of the benefits of centralised planning with tactical adjustments to new information by subordinate units. However, this architecture is not guaranteed to produce an optimal solution if tempo is high, since the delays inherent in the hierarchy may negate the benefits of centralised planning. For example, in the Gulf War ground campaign, Norman Schwarzkopf acted as both CINC and ground component commander, and there were two levels of command (3rd Army and VII and XVIII Corps) between him and the US Army divisions on the ground, although a single level may have been more appropriate. General (Ret) Fred

Franks (who commanded VII Corps) in his assisted autobiography [12] records a number of ways in which the multiple levels of hierarchy caused delays and misunderstandings which partially compromised the success of this high-tempo campaign. In the words of Clausewitz, cited in [16]:

"There is no denying that the supreme command of an army... is markedly simpler if orders only need to be given to three or four other men; yet a general has to pay dearly for that convenience... an order progressively loses speed, vigor and precision the longer the chain of command it has to travel, which is the case where there are corps commanders placed between the divisional commanders and the general."

It should be pointed out, however, that our experiment does not entirely do justice to this model because our task is too simple to benefit from being hierarchically decomposed: it can be effectively solved centrally. Our experiment also does not allow for tactical adjustments to a central plan, and in future work we intend to address this.

2.3 The Distributed Architecture Without Information Sharing

In the distributed architecture (see figure 3), there are four independent commands. Each command "owns" a column of the grid, with one surveillance aircraft and one fighter squadron dedicated to that column. Each independent command also receives satellite intelligence. Due to the short sensor-to-shooter pathway and the reduced need for coordination, with this architecture air strikes occur only 3 time steps after intelligence collection begins. However, it is possible for one column to have many targets, only one of which receives an air strike, while another column has its fighter squadron unused.

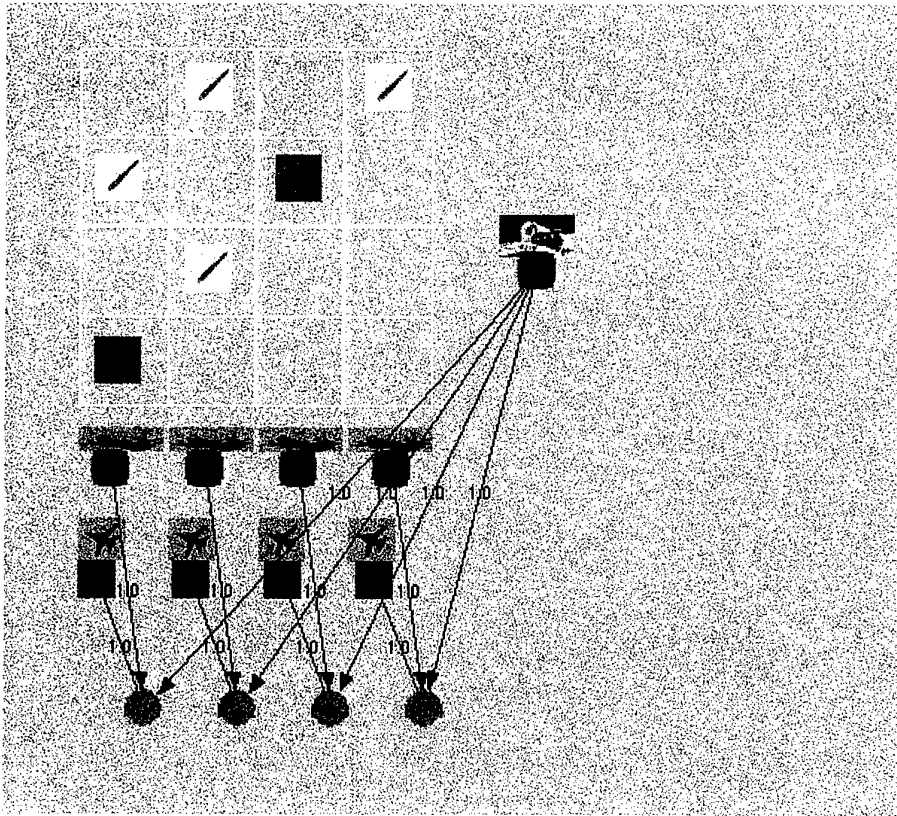


Figure 3: Distributed Architecture without Information Sharing

The distributed architecture is the traditional model of special operations, where communications are poor or non-existent, and lethal events which can terminate the mission can occur on a minute-by-minute basis. In such an environment, it is appropriate for units on the ground to make decisions without reference to higher command.

2.4 The Negotiation Architecture Without Information Sharing

The negotiation architecture (see figure 4), is similar to the distributed architecture, but each independent command can pass excess targets in its column to the commands on its immediate left and right. This produces a result intermediate between the centralised and distributed architectures, and air strikes with this architecture occur 4 time steps after intelligence collection begins.

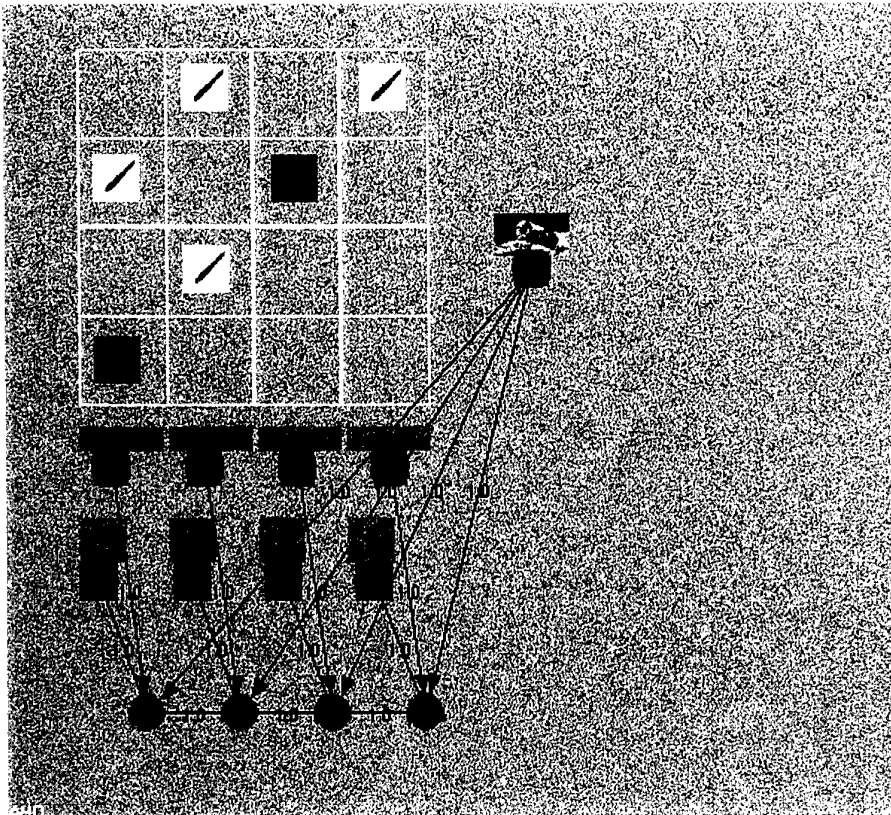


Figure 4: Negotiation Architecture without Information Sharing

The negotiation architecture (also known as peer-to-peer) is the traditional model of emergency services (fire and ambulance) where each station is independently responsible for a fixed area. This is appropriate, since for emergency services speed of response is essential above all. Should a station be overloaded with multiple emergencies, it can negotiate with adjacent stations to handle the excess calls. Only for major (city-wide or state-wide) emergencies is a centralised command structure put into place.

2.5 Architectures with Information Sharing

Each of the above architectures has an information sharing variant, in which the surveillance aircraft are upgraded to provide intelligence not only on the column along which they fly, but also on the immediately adjacent columns. For the centralised and split architectures, this additional intelligence is handled by the intelligence headquarters. For the distributed (see figure 5) and negotiation (see figure 6) architectures, this requires additional communication pathways from the surveillance aircraft to adjacent commands. In either case, the additional information is obtained at a cost of 1 additional time step.

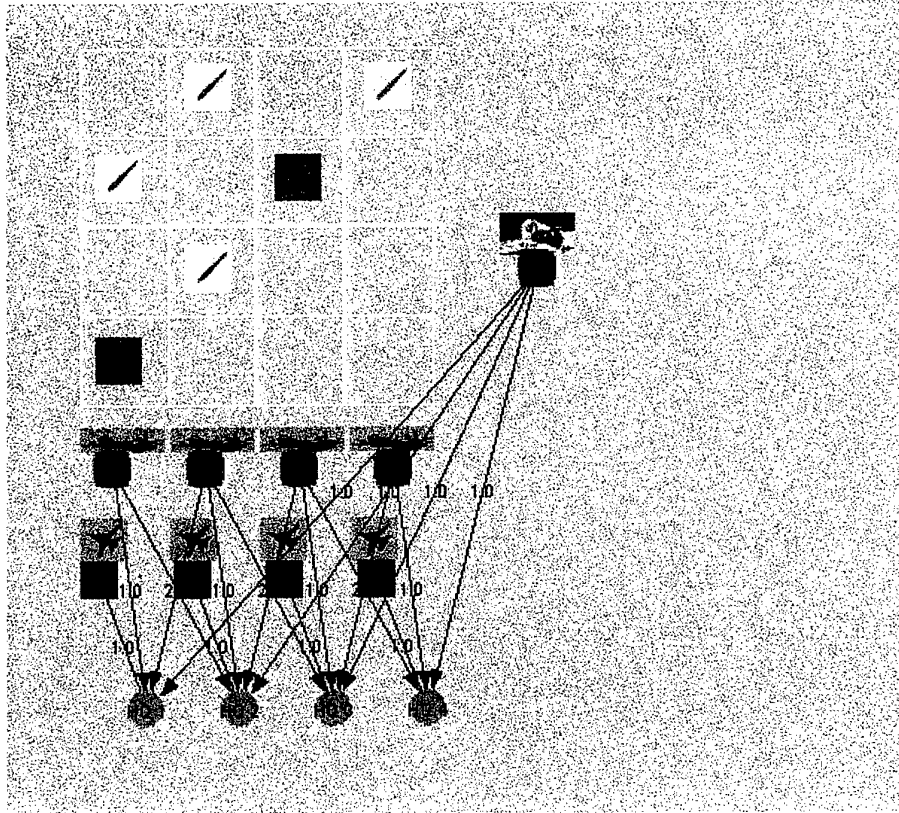


Figure 5: Distributed Architecture with Information Sharing

The negotiation architecture with information sharing represents the emerging paradigm of Network Centric Warfare [13], where information is shared through a network of sensors. Network Centric Warfare (NCW) realises its full potential where units can negotiate tasks with each other (self-synchronisation) in response to rapidly changing situations without first contacting higher command. In the words of David Alberts et al [13]:

*"NCW offers the opportunity not only to be able to develop and execute highly synchronized operations, but also to explore C2 approaches based upon horizontal coordination, or self-synchronization, of actor entities. In fact, the Marines have adopted **Command and Coordination** as their preferred term for command and control in future operations."*

Table 1 summarises the eight architectures and their associated time delays.

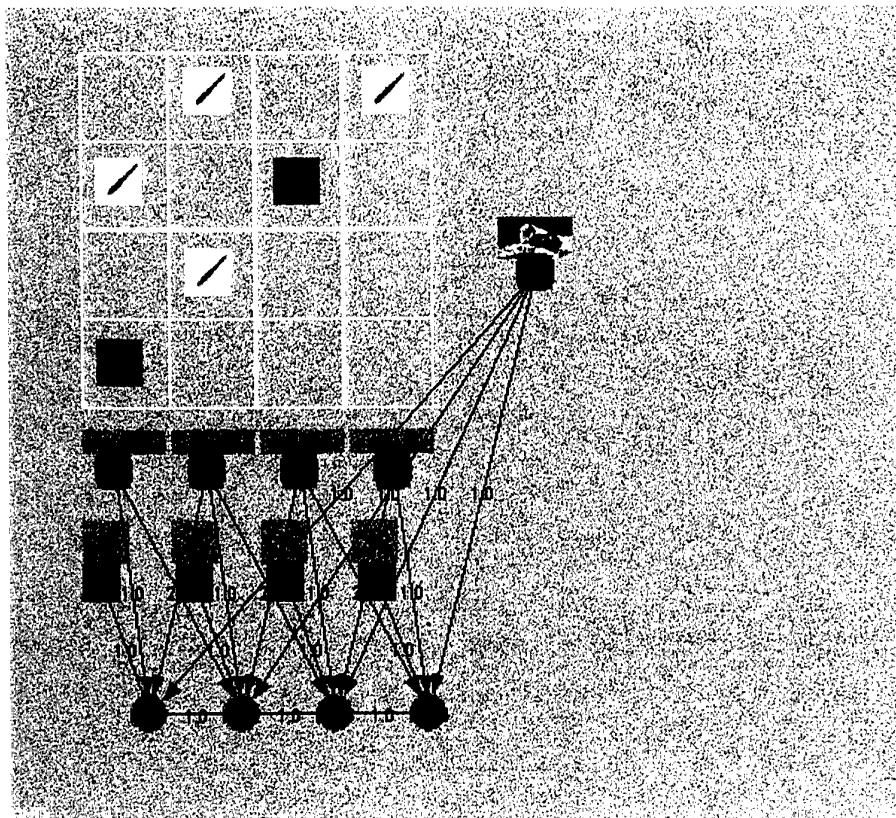


Figure 6: Negotiation Architecture with Information Sharing

Table 1: Eight C4ISR Architectures and Associated Time Delays

	No Information Sharing	Information Sharing
Centralised	5	6
Distributed	3	4
Negotiation	4	5
Split	6	7

3. Experimental Results

Table 8 in the appendix shows the number of missile launch sites destroyed for each of the eight architectures, under various combinations of tempo p and sensor quality q . Each number in the table is the average of 10,000 experimental runs. The best performance for each combination of p and q is indicated by an asterisk.

Table 2 shows the average number of missile launch sites destroyed for each of the eight architectures (averaged over all combinations of p and q examined). The overall best-performing architecture is the negotiation architecture with information sharing. The centralised architecture with information sharing and the negotiation architecture without information sharing also perform better than the overall average of 1.503 targets destroyed.

Table 2: Average Performance for Eight Architectures

Overall Average	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
1.503	1.491	1.416	1.562	1.339	1.625	1.455	* 1.689	1.448

Table 3 shows the best-performing architecture for each combination of tempo p and sensor quality q . It can be seen that the performance of each architecture depends very much on tempo and sensor quality. The overall best-performing architecture (negotiation with information sharing) is highlighted, and it can be seen that this architecture is only best for certain combinations of tempo and sensor quality. It can also be seen that the transition from distributed architecture being best to centralised architecture being best (whether vertical or horizontal) always has an intermediate stage where the negotiation architecture is best. However, before we discuss our experimental results further, we will introduce the FINC methodology.

Table 3: Best-Performing Architectures for Tempo and Sensor Quality Combinations

Prob.	$q = 0.1$	$q = 0.25$	$q = 0.5$	$q = 0.75$	$q = 0.9$
0.01	Distr. Share	Negot. Share	Centr. Share	Centr. Share	Centr. Share
0.02	Distr. Share	Negot. Share	Negot. Share	Centr. Share	Centr. Share
0.05	Distr. Share	Negot. Share	Negot. Share	Negot. Share	Negotiation
0.1	Distr. Share	Distr. Share	Negot. Share	Negot. Share	Negotiation
0.2	Distributed	Distributed	Distributed	Negotiation	Negotiation
0.5	Distributed	Distributed	Distributed	Distributed	Distributed

4. The FINC (Force, Intelligence, Networking and C2) Methodology

In this section we describe the FINC methodology [3], using a simple military structure (figure 7) as an illustrative example. In this example, two brigade-level units (BDE 1 and BDE 2) are controlled by a divisional-level headquarters (DIV HQ), which in turn is controlled by a joint headquarters (JNT HQ) which also controls strategic intelligence and air assets. We provide this example structure purely in order to describe the FINC methodology, and are not suggesting that it is appropriate for any specific purpose. In the next section of this paper we provide an application of the FINC methodology to our "SCUD hunt" experiment.

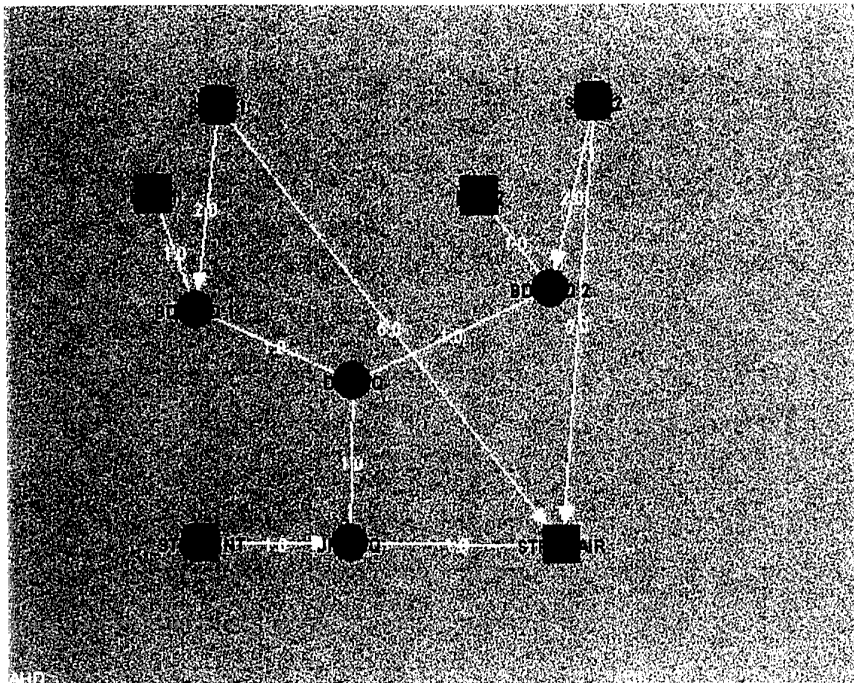


Figure 7: A Simple Military Organisational Structure

The FINC methodology analyses an organisational structure relatively simply in terms of **force, intelligence, networking, and C2** assets. Force assets are those which carry out any kind of military task, and are indicated by square boxes in figure 7. Intelligence assets collect any kind of information, and are indicated by rounded boxes in figure 7. Networking provides communication between assets, indicated by lines or arrows in figure 7 (depending on whether information flow is unidirectional or bidirectional). C2 (command and control) assets make decisions, and are indicated by circles in figure 7. The force and intelligence assets are often themselves organisations that can be subdivided in a similar way, if necessary. Having divided an organisation in this way, the FINC methodology provides a number of **metrics**, for evaluating the efficiency of the organisational structure.

The FINC methodology is also applicable to business organisations. For example, the force assets which carry out tasks could include the sales force and business units; intelligence assets could include research and development, market research,

and recorded sales figures; and C2 assets could include management and decision-makers.

Each force and intelligence asset has an associated **area of operations**, which for simplicity is assumed to be approximately circular. In figure 7 these assets are:

```
Scout unit 1 (Intelligence), radius = 100 (in arbitrary units)
Scout unit 2 (Intelligence), radius = 100
Brigade BDE 1 (Force), radius = 100
Brigade BDE 2 (Force), radius = 100
Strategic air (STRAT AIR) assets (Force), radius = 400
Strategic intelligence (STRAT INT) assets (Intelligence), radius = 400
```

In cases where the areas of operation for intelligence and force assets overlap, there is benefit in providing a flow of information from the intelligence asset to the force asset. In figure 7, candidate information flows are:

```
Scout unit 1 to Brigade BDE 1
Scout unit 2 to Brigade BDE 2
Strategic intelligence (STRAT INT) to Brigade BDE 1
Strategic intelligence (STRAT INT) to Brigade BDE 2

Scout unit 1 to Strategic air (STRAT AIR)
Scout unit 2 to Strategic air (STRAT AIR)
Strategic intelligence (STRAT INT) to Strategic air (STRAT AIR)
```

Note that intelligence from Scout units is only useful to their associated Brigade units, since in this example, the areas of operation of the two brigades do not overlap. However, the strategic intelligence and air assets (with radius = 400) overlap with both brigades.

Intelligence assets differ in the kind of information they provide. Although such differences can be quite complex, for simplicity we model this by associating a **mode** or **band** with each intelligence asset. If a single asset produces different kinds of information, we simply model it as multiple co-located assets. We assume that two intelligence assets in different bands are complementary, while intelligence assets in the same band provide duplicate information. Intelligence assets also differ in the **quality** of information they provide. We model this using a numerical quality score for each intelligence asset. Given two intelligence assets in the same band, we prefer the highest quality information, and can discard the lower quality information. For figure 7, intelligence assets are assumed to be in the same band, and quality (in arbitrary units) is taken to be:

```
Scout unit 1 (Intelligence), quality = 0.5
Scout unit 2 (Intelligence), quality = 0.5
Strategic intelligence (STRAT INT) assets (Intelligence) quality = 0.2
```

In other words, the strategic intelligence assets in this example provide information which overlaps with the information provided by scout units, and which is lower-quality but available over a wider area (we emphasise that this example is not realistic, and is provided merely to illustrate the methodology). The issue of how actual sensor characteristics are translated to numerical quality scores is outside the scope of the present paper, and we intend to address this in future work.

Each communication link in the network has varying reliability and bandwidth characteristics which for simplicity we model as an average **delay factor** for the transfer of information across the link. The key idea here is not the message transmission time, but the time to get across an understanding of reports or instructions. This may require multiple exchanges and clearly takes longer with low-bandwidth communication while face-to-face communication reaches understanding more rapidly. Delays (in arbitrary units) are indicated on the links in figure 7. For the experiment we describe in this paper, the delay factors are estimated (based on the description of the experimental testbed in the previous section), but for real-world studies we calculate the delay using a formula which is simple, but still of value in predicting performance:

$$\text{delay factor} = \text{actual delay} * \text{misunderstanding factor} / \text{amount of information}$$

Here the actual delay is the time required to actually send the block of information, i.e. the transmission time plus the average time between transmissions. This will depend on communications bandwidth, availability of the communications technology involved, set-up time, and standard operating procedures. The amount of information per transmission is measured in bytes (assuming the best possible compression technology is used). The misunderstanding factor is usually taken to be 1.0, but for organisations which involve multiple cultures, the misunderstanding factor will be greater than 1.0 for cross-cultural links. Such cross-cultural links include communication between different services (such as between the US Army, Air Force, and Marines in the Gulf War [11, 12]), or communication between units from different countries. Further work is still needed to assess the suitability of this calculation of the delay factor.

Each C2 node in the architecture processes intelligence information and passes it on (as well as many other C2 functions). This introduces an additional delay factor which is added to the delay factor for communication links. In figure 7, all delays for C2 nodes are assumed to be 1.0 (in the same arbitrary units as for links).

The FINC methodology uses the information in this model to conduct three kinds of analysis: **delay analysis**, **centrality analysis**, and **intelligence analysis**.

4.1 Delay Analysis 1: the information flow coefficient

In delay analysis, we consider the combined delay (i.e. the combination of communication delays and C2 delays) for each candidate information flow. Where multiple communication paths exist, we take the one with the shortest delay. For figure 7, the delays for the candidate information flows are:

Scout unit 1 to Brigade BDE 1, delay = 2.0 + 1.0 + 1.0 = 4.0
 Scout unit 2 to Brigade BDE 2, delay = 2.0 + 1.0 + 1.0 = 4.0
 Strategic intelligence (STRAT INT) to Brigade BDE 1, delay = 7.0
 Strategic intelligence (STRAT INT) to Brigade BDE 2, delay = 7.0

Scout unit 1 to Strategic air (STRAT AIR), delay = 5.0
 Scout unit 2 to Strategic air (STRAT AIR), delay = 5.0
 Strategic intelligence (STRAT INT) to Strategic air (STRAT AIR), delay = 3.0

The first metric we use for assessing C4ISR architectures is simply the average of these delay values, which we call the **information flow coefficient**. It provides a measure of how effectively the military organisation can mobilise information to carry out a task. For the example in figure 7, this coefficient is 5.0. For this metric, low values are desirable.

The information flow coefficient provides one simple way of assessing changes to the military structure. For example, eliminating the direct links between scout units and strategic air assets in figure 7 reduces the effectiveness of information flow, and increases the information flow coefficient to 5.86. Since it measures the delay between obtaining information and acting on it, the information flow coefficient provides an indication of **tempo superiority**, i.e. the ability to react more rapidly than an adversary.

4.2 Delay Analysis 2: the coordination coefficient

The second metric we use for assessing C4ISR architectures is the **coordination coefficient**. It provides a measure of how effectively the military organisation can coordinate activities. This metric is calculated by averaging the delays along paths connecting force assets. This is very similar to the information flow coefficient, but the information flow coefficient considers paths from relevant intelligence assets to force assets, while the coordination coefficient considers paths between force assets. For the example in figure 7, these paths are:

Brigade BDE 1 to Brigade BDE 2 and vice versa, delay = 7.0
 Brigade BDE 1 to Strategic air (STRAT AIR) and vice versa, delay = 7.0
 Brigade BDE 2 to Strategic air (STRAT AIR) and vice versa, delay = 7.0

Consequently, the coordination coefficient is 7.0. For this metric, low values are also desirable. It provides an indication of **coordination superiority**, i.e. the ability to orchestrate (in the words of General Sir John Monash) multiple actions more effectively than an adversary.

4.3 Centrality Analysis

In centrality analysis, we try to identify the most "central" node in the architecture, which provides some indication of the "centre of gravity" [14] of the structure. Centrality is a traditional idea in Social Network Analysis, and there are several possible definitions of the concept [1], but a suitable definition for the degree of centrality of node i in a network where there is a concept of varying "distance" or "strength" of links is:

$$\frac{1}{2} \text{ AVERAGE } (j \neq i) \{1 / \text{delay}(i, j)\} + \frac{1}{2} \text{ AVERAGE } (j \neq i) \{1 / \text{delay}(j, i)\}$$

i.e. the centrality score for a particular node is the sum of inverse distances to all the other nodes — the most central node is the one that is "closest" to everything else. Node that in general the delay from i to j may be different from the reverse delay. For the example in figure 7, the delay from STRAT INT to STRAT AIR is 2.0, but the reverse delay is infinite, since the link from STRAT INT to JNT HQ is unidirectional.

For the network in figure 7, the most central node is the divisional headquarters (DIV HQ), while the second most central node is the joint headquarters (JNT HQ). This provides an indication that the architecture in figure 7 is indeed a land-focused rather than a joint-focused structure.

Note that we do not use centrality analysis in analysing our "SCUD hunt" experiment.

4.4 Intelligence Analysis: the intelligence coefficient

Our third form of analysis measures the degree to which intelligence is used. For each candidate information flow from an intelligence asset to a force asset (which uses the intelligence), we estimate the **effective intelligence quality** to be the intelligence quality discussed above divided by the delay factor for the path, to allow for the decrease in value of information as it ages. This is a somewhat crude calculation, since some information retains its value even after considerable time has passed, while other information becomes useless almost immediately. However, this calculation provides a simple approximation to the way that information loses value over time. For the example in figure 7 we have:

```
Scout unit 1 to BDE 1, delay = 4.0, quality = 0.5, effective quality = 0.125
Scout unit 2 to BDE 2, delay = 4.0, quality = 0.5, effective quality = 0.125
STRAT INT to BDE 1, delay = 7.0, quality = 0.2, effective quality = 0.029
STRAT INT to BDE 2, delay = 7.0, quality = 0.2, effective quality = 0.029

Scout unit 1 to STRAT AIR, delay = 5.0, quality = 0.5, effective quality =
0.1
Scout unit 2 to STRAT AIR, delay = 5.0, quality = 0.5, effective quality =
0.1
STRAT INT to STRAT AIR, delay = 3.0, quality = 0.2, effective quality =
0.067
```

These calculations are repeated for each intelligence band or mode.

For each force asset and intelligence band, we calculate an **intelligence volume** which is the product of effective intelligence quality and relative area (within the area of operations of the force asset) covered by the intelligence information. In cases where the areas of operations of intelligence and force assets only partially overlap, we assume that there is sufficient flexibility of position to make this overlap total when needed.

For example, for the strategic air (STRAT AIR) asset in figure 7, strategic intelligence covers the entire area of operations (radius = 400) with effective intelligence quality = 0.067, while the two scout units cover smaller areas (radius = 100) with slightly higher effective intelligence quality = 0.1 of the same kind of information. Figure 8 illustrates this:

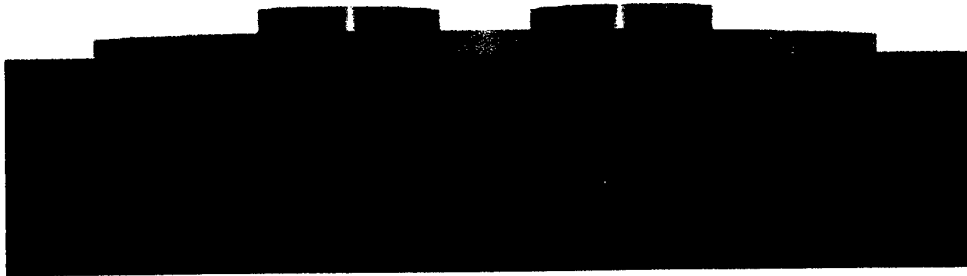


Figure 8: Intelligence Volume for Strategic Air Asset

In this diagram, the intelligence assets relevant to STRAT AIR are indicated by transparent green cylinders. The area of each cylinder indicates the physical area covered by the intelligence asset. The height of each cylinder indicates the corresponding effective intelligence quality, so that the two cylinders representing scout units stand out above the slightly lower effective intelligence quality of the strategic intelligence (STRAT INT) asset. The **intelligence volume** for the strategic air asset is simply the total volume of the combined shape (divided by pi for simplicity):

$$\begin{aligned}
 &\text{intelligence volume for STRAT AIR} \\
 &= 0.067 * 400 * 400 + (0.1 - 0.067) * 100 * 100 + (0.1 - 0.067) * 100 * 100 \\
 &= 10720 + 330 + 330 \\
 &= 11380
 \end{aligned}$$

The intelligence volume for each brigade ignores strategic intelligence assets, since for this example we assume that the scout units provide exactly the same kind of intelligence and they have a higher effective intelligence quality of 0.125:

$$\begin{aligned}
 &\text{intelligence volume for BDE 1 or BDE 2} \\
 &= 0.125 * 100 * 100 \\
 &= 1250
 \end{aligned}$$

The **intelligence coefficient** of the architecture is simply the total of the intelligence volumes for each force asset and intelligence band. For figure 7 this is $11380 + 1250 + 1250 = 13800$, approximately. For this metric, large values are desirable.

The intelligence coefficient can be improved either by improving the quality of individual intelligence assets, decreasing the delay on communication paths, or by adding intelligence assets (on new bands) which complement existing assets. We believe this metric provides a reasonable way of assessing the impact of such changes. Essentially this metric provides an indication of **information superiority**, i.e. the ability to obtain and utilise information more effectively than an adversary.

5. Modelling the Testbed with the FINC Methodology

For the purposes of the FINC methodology, we model our eight experimental architectures with a delay factor of 1 for each headquarters and link, unless otherwise specified. This is sufficient to describe the centralised, distributed, and split architectures without information sharing. The link delays (but not the headquarters delays) are shown in figures 1 through 6. For the negotiation architectures, we need to model the extra delay due to the negotiation process. We do this by adding an extra 0.5 to the delays on each headquarters, which adds 1 to the total path delays between adjacent squadrons.

For the information sharing architectures, we need to model the extra delay due to processing the additional intelligence. For the centralised and split architectures we can do this easily by increasing the delay on the intelligence headquarters from 1 to 2 to allow for the increased workload. For the distributed and negotiation architectures (which have no intelligence headquarters) we model the delays associated with information sharing by placing a delay factor of 2 on the new intelligence links to adjacent columns plus an extra delay of 0.5 on each headquarters.

The process of modelling the eight experimental architectures by choosing delay values for each headquarters and link (which we have just described) has been largely based on estimation. For a real-life scenario, a similar estimation process would be necessary. However, once the estimated delay values have been chosen, we can calculate the information flow coefficient, coordination coefficient and intelligence coefficient using the procedures outlined in the previous section. For example, for the centralised architecture without information sharing, we have the following candidate information flows and total path delays:

```

Surveillance plane 1 to Squadron 1, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Surveillance plane 2 to Squadron 2, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Surveillance plane 3 to Squadron 3, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Surveillance plane 4 to Squadron 4, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Satellite to Squadron 1, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Satellite to Squadron 2, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Satellite to Squadron 3, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0
Satellite to Squadron 4, delay = 1.0 + 1.0 + 1.0 + 1.0 + 1.0 = 5.0

```

The information flow coefficient is the average of these (identical) delay values, which is 5.0. Table 4 shows the information flow coefficient (*info*), the coordination coefficient (*coord*), and the intelligence coefficient (*intel*) calculated in a similar way for each of the eight experimental C4ISR architectures (the actual calculations were performed by our CAVALIER tool). Figure 9 shows the same information graphically. Calculation of the intelligence coefficient requires modelling sensor quality. We do this using the sensor quality discussed when the testbed was described, so that the intelligence coefficient is a multiple of q . Since the various sensors are all complementary, we model them as five distinct intelligence bands.

The coordination coefficient for the distributed architectures turns out to be infinite (since there is no path between different squadrons), but we substitute the number 15 for the purpose of statistical analysis.

Table 4: Eight C4ISR Architectures and Results of FINC Modelling

	information flow coefficient (<i>info</i>)	coordination coefficient (<i>coord</i>)	intelligence coefficient (<i>intel</i>)
Centralised	5	3	16000 <i>q</i>
Distributed	3	Infinity (15)	26667 <i>q</i>
Negotiation	3.5	7.667	22857 <i>q</i>
Split	7	5.667	11429 <i>q</i>
Centralised Sharing	6	3	23333 <i>q</i>
Distributed Sharing	3.929	Infinity (15)	36190 <i>q</i>
Negotiation Sharing	4.429	9	32000 <i>q</i>
Split Sharing	8	5.667	17500 <i>q</i>

Comparing the information flow coefficients in table 4 with the actual time delays between intelligence collection and air strike (shown in table 1), we can see that they are similar but not precisely identical. This is what we would expect in reality. In general, the three coefficients calculated by the FINC methodology (being based on estimation process) will be a reflection of reality, but not a perfect one. However, although the FINC coefficients are imperfect, regression analysis shows that they predict performance extremely well (at least for this "SCUD hunt" scenario).



Figure 9: 3-Dimensional Graph of Results of FINC Modelling for Eight C4ISR Architectures

Regression analysis produces the following equation of best fit for the number of targets destroyed, in terms of the three FINC metrics, and the probability p which provides an indication of battlefield tempo:

$$0.287 - 8.685 p + 11.992 p^2 + 0.02566 \sqrt{\text{intel}} - 0.000003703 \text{ intel-coord} - 0.00003692 \text{ intel-info-p}$$

Each parameter here is significant at the $P < 0.0001$ level under two-tailed t-tests, and the equation explains 94% of the variance in the experimental data in table 8 (in other words, the correlation is 0.97). This is an extremely good result, and is limited only by the fact that the equation does not fully capture the complex nonlinearities in the data. We provide a more detailed explanation of the equation of best fit below.

Table 9 in the appendix shows the number of targets destroyed estimated using the equation of best fit. It can be seen that the results are not perfect, since the equation does not fully capture all the nonlinearities, but for the 30 combinations of p and q , the equation of best fit identifies the best architecture on 12 occasions, the second or third best architecture on 14 occasions, and the fourth best architecture on the remaining 4 occasions. This provides another indication that the equation of best fit is a good one, but the true utility of the equation lies in the principles of C4ISR architectures that it embodies, and we address this next, by examining in detail the meaning of the different parts of the best-fit equation:

5.1 Explaining the Equation of Best Fit (a): $0.287 - 8.685 p + 11.992 p^2$

Inspection of the data in table 8 shows that the biggest impact on the number of targets destroyed is battlefield tempo, measured by the probability p . The relationship between p and the number of targets destroyed is a nonlinear one, and so the equation of best fit has a quadratic dependence on p . This explains 61% of the variance in the data (a linear dependence on p would explain only 55%).

Including cubic or higher powers of p does not explain more of the variance, so that the remaining 39% of the variance is due to sensor quality and the choice of architecture.

5.2 Explaining the Equation of Best Fit (b): $0.02566 \sqrt{\text{intel}}$

The second largest impact on the number of targets destroyed is the intelligence coefficient (which essentially represents an estimate of the total amount of information discounted by its age). This explains an additional 23% of the variance (for a total of 84%). The relationship between the intelligence coefficient and the number of targets destroyed is a nonlinear one, and the square root of the intelligence coefficient provides a better indication than the intelligence coefficient itself (the intelligence coefficient itself would explain only an additional 20% of the variance).

The significance of this factor indicates the importance of having a high intelligence coefficient, or in other words, having information superiority.

5.3 Explaining the Equation of Best Fit (c): $-0.000003703 \text{ intel} \cdot \text{coord}$

The next factor influencing the number of targets destroyed is the product of the intelligence coefficient and the coordination coefficient, which explains an additional 4% of the variance (for a total of 88%). The coordination coefficient itself provides no additional explanatory power.

The negative coefficient for this factor in the equation indicates the importance of balancing a high intelligence coefficient (information superiority) with a low coordination coefficient (coordination superiority). The product relationship indicates that with low intelligence coefficients (i.e. when sensor quality q is low), the need to make the greatest possible use of limited intelligence is paramount (and the coordination coefficient is less important). However with higher intelligence coefficients (i.e. when sensor quality q is moderate to high), the need for coordination (i.e. a low coordination coefficient) becomes more important.

5.4 Explaining the Equation of Best Fit (d): $-0.00003692 \text{ intel} \cdot \text{info} \cdot p$

The final factor influencing the number of targets destroyed is the product of the intelligence coefficient, the information flow coefficient, and the probability p . This explains an additional 6% of the variance (for a grand total of 94%). The information flow coefficient itself provides very little additional explanatory power (less than 1%), and is not included in the equation.

The negative coefficient for this factor in the equation indicates the importance of balancing a high intelligence coefficient (information superiority) with a low information flow coefficient (speed, or tempo superiority) when the tempo p is high.

6. Four Basic Principles of C4ISR Architectures

Table 5 shows a simplified view of the best-performing architectures in table 3. It must be remembered that the transition from distributed architecture being best to centralised architecture being best has an intermediate stage where the negotiation architecture is best. This table is clearly subdivided into four regions, as shown in table 6.

Table 5: Best-Performing Architectures (Simplified)

	Poor sensors ($q=0.1$)	Fair sensors ($q=0.5$)	Good sensors ($q=0.9$)
Slow tempo ($p = 0.01$)	Distr. Share	Centr. Share	Centr. Share
Moderate tempo ($p = 0.05/0.1$)	Distr. Share	Negot. Share	Negotiation
Fast tempo ($p = 0.5$)	Distributed	Distributed	Distributed

Each of the four performance regions displays a different combination of the factors which we identified in the equation of best fit.

Table 6: Performance Regions

	Poor sensors ($q=0.1$)	Fair to Good sensors ($q=0.5 / 0.9$)
Slow tempo ($p = 0.01$)	A	B
Moderate tempo ($p = 0.05/0.1$)		C
Fast tempo ($p = 0.5$)	D	

6.1 Region A: Slow to Moderate Tempo and Poor Sensors — Intelligence Superiority Critical

This region is dominated by the need to make the greatest possible use of limited intelligence (i.e. to maintain the highest possible intelligence coefficient in the face of poor sensor quality), and it is in this region that the distributed sharing architecture (the one with the highest intelligence coefficient) performs best.

6.2 Region B: Slow Tempo and Fair to Good Sensors — Intelligence & Coordination Superiority

This region is dominated by the balance between a reasonably high intelligence coefficient and a very low coordination coefficient (i.e. between intelligence superiority and coordination superiority), as described in the discussion of the *intel · coord* factor.

It is in this region that the centralised sharing architecture (the one with the fourth highest intelligence coefficient and the lowest coordination coefficient) performs best, and this appears to be the region in which the US Air Force operates, with its centralised Air Tasking Orders (in this case, tempo is only “slow” in comparison to the US Air Force’s extremely rapid ability to respond).

6.3 Region C: Moderate Tempo and Fair to Good Sensors — Combined Superiority

This region is dominated by the balance between a reasonably high intelligence coefficient and both low coordination coefficients and low information flow coefficients (i.e. a three-way balance between intelligence superiority, coordination superiority and tempo superiority), as described in the discussion of the *intel · coord* and *intel · info · p* factors. At the low-sensor-quality end of this region the intelligence coefficient is slightly more important.

It is in this region that the negotiation architectures performs best (with sharing when necessary, and without sharing when sensors are good). The negotiation

architecture with sharing balances all three factors with a slight bias towards the intelligence coefficient, i.e. information superiority (it has the fourth-lowest information flow coefficient, the sixth-lowest coordination coefficient, and the second-highest intelligence coefficient). The negotiation architecture without sharing balances all three factors with a slight bias towards the information flow coefficient, i.e. tempo superiority (it has the second-lowest information flow coefficient, the fifth-lowest coordination coefficient and the fifth-highest intelligence coefficient). In other words, the negotiation architectures seem to be the most successful at balancing the three kinds of superiority, approaching the global optimum produced by the centralised architecture while providing a more rapid response.

6.4 Region D: Fast Tempo — Intelligence & Tempo Superiority

This region is dominated by the balance between a reasonably high intelligence coefficient and a very low information flow coefficient (i.e. between intelligence superiority and tempo superiority), as described in the discussion of the *intel · info · p* factor.

It is in this region that the distributed architecture without sharing (the one with the third highest intelligence coefficient and the lowest information flow coefficient) performs best, and this appears to be the region in which Special Forces operate.

The existence of this region is consistent with the results of chess-based experiments conducted by the Swedish National Defence College [15] which indicate that there are environments in which tempo superiority becomes more important than information superiority. By our standards, chess is a high-tempo game, since the situation can change radically in a single move.

Table 7 summarises the principles of C4ISR architectures demonstrated in these four regions.

Table 7: Principles of C4ISR Architectures

	Poor sensors	Fair to Good sensors
Slow tempo	Region A: <i>Information superiority</i> is most important (high intelligence coefficient)	Region B: Balance <i>information superiority</i> and <i>coordination superiority</i> (high intelligence coefficient and low coordination coefficient)
Moderate tempo		Region C: Balance <i>all three kinds</i> of superiority (high intelligence coefficient, low information flow coefficient, and low coordination coefficient)
Fast tempo		Region D: Balance <i>information superiority</i> and <i>tempo superiority</i> (high intelligence coefficient and low information flow coefficient)

7. Conclusions

We have presented a methodology for evaluating and comparing organisational structures which we call FINC (Force, Intelligence, Networking and C2), and a very simple testbed for evaluating organisational structures. Our experiment has validated the FINC methodology, since performance is successfully predicted by the three FINC metrics (the information flow coefficient measuring tempo superiority, the coordination coefficient measuring coordination superiority, and the intelligence coefficient measuring information superiority).

Analysis of the experimental results identifies four regions (A, B, C, and D) in the tempo/sensor quality space in which different kinds of superiority are important. This highlights the ways in which different military organisations — air, sea, and land — have successfully adapted to suit their intelligence environment and tempo. However, as information environments and tempo change, traditional organisational structures may no longer be appropriate.

The first task in designing new C4ISR architectures is therefore to ask: in which of the four regions (A, B, C, or D) will the architecture be operating?

Further work is still required on FINC methodology, specifically on the process for the calculating delay factors for headquarters and links, and for assigning quality scores to intelligence assets. We also intend to further validate the FINC methodology by applying to other simulation experiments and to real-world examples.

8. Acknowledgements

The author is indebted to Moira Chin, Gina Kingston, Toby Keene, and Pin Chen for useful discussions on C4ISR architectures, and to Jon Rigter, Kim Blackmore, and Ed Kruzins for many insightful comments on earlier drafts of this paper.

9. References

- [1] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*, Cambridge University Press, 1994.
- [2] Anthony H. Dekker. "Social Network Analysis in Military Headquarters using CAVALIER," *Proceedings of 5th International Command and Control Research and Technology Symposium*, Australian War Memorial, Canberra ACT, Australia, 24-26 October 2000. The full text of the paper is available electronically at <http://www.dodccrp.org/2000ICCRTS/cd/papers/Track6/039.pdf>

- [3] Anthony H. Dekker. "Applying Social Network Analysis Concepts to Military C4ISR Architectures," to appear in *Connections, the official journal of the International Network for Social Network Analysis*.
- [4] Jonathan Weiner. *Time, Love, Memory: A Great Biologist and His Quest for the Origins of Behaviour*, Faber and Faber, 1999.
- [5] Kathleen M. Carley and Michael J. Prietula. "ACTS Theory: Extending the Model of Bounded Rationality," *Computational Organization Theory*, pp 55-87, Kathleen M. Carley and Michael J. Prietula eds, Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.
- [6] Zhiang Lin. "A Theoretical Evaluation of Measures of Organizational Design: Interrelationship and Performance Predictability," *Computational Organization Theory*, pp 113-159, Kathleen M. Carley and Michael J. Prietula eds, Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.
- [7] Julia Loughran. "Working Together Virtually: The Care and Feeding of Global Virtual Teams," *Proceedings of 5th International Command and Control Research and Technology Symposium*, Australian War Memorial, Canberra ACT, Australia, 24-26 October 2000. The full text of the paper is available electronically at <http://www.dodccrp.org/2000ICCRTS/cd/papers/Track4/009.pdf> — see also <http://www.scudhunt.com/>
- [8] C4ISR Architecture Working Group, (US) Department of Defense. *C4ISR Architecture Framework Version 2.0*.
- [9] Alan Gibbons. *Algorithmic Graph Theory*, Cambridge University Press, 1985.
- [10] David Krackhardt. "Graph Theoretical Dimensions of Informal Organizations," *Computational Organization Theory*, pp 89-111, Kathleen M. Carley and Michael J. Prietula eds, Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.
- [11] Tom Clancy and General Chuck Horner (Ret), *Every Man a Tiger: The Gulf War Air Campaign*, Berkley Books, 1999.
- [12] Tom Clancy and General Fred Franks Jr (Ret), *Into the Storm: A Study in Command*, Sedgwick and Jackson, 1999.
- [13] David S. Alberts, John J. Garstka and Frederick P. Stein. *Network Centric Warfare: Developing and Leveraging Information Superiority*, 2nd ed, (US) Department of Defense C4ISR Cooperative Research Program Publications Series, 1999. Available electronically at http://www.dodccrp.org/Publications/pdf/ncw_2nd.pdf
- [14] Carl von Clausewitz (tr. J. J. Graham and F. N. Maude), *On War*, Wordsworth, 1997. See http://www.clausewitz.com/CWZHOME/On_War/ONWARTOC.html
- [15] Jan Kuylenstierna, Joacim Rydmark and Tonie Fahraeus. "The Value of Information in War: Some Experimental Findings," *Proceedings of 5th International Command and Control Research and Technology Symposium*, Australian War Memorial, Canberra ACT, Australia, 24-26 October 2000. The full text of the paper is available electronically at <http://www.dodccrp.org/2000ICCRTS/cd/papers/Track1/014.pdf>

- [16] Kenneth Allard, *Command, Control, and the Common Defense*, Revised edition, (US) National Defense University Press, 1996.
- [17] Andrew W. Gill, Richard Egudo, Peter J. Dortmans and Dion Grieger. "Supporting the Army Capability Development Process Using Agent-Based Distillations: A Case Study," *Journal of Battlefield Technology*, Vol 4, No 3, November 2001, pp 24-29.
- [18] Anthony H. Dekker. "Visualisation of Social Networks using CAVALIER," *Proceedings of the Australian Symposium on Information Visualisation*, Sydney, Australia, 3-4 December 2001, pp 49-55. Conferences in Research and Practice in Information Technology, Vol. 9, Peter Eades and Tim Pattison, eds. Available electronically at <http://www.jrpit.flinders.edu.au/confpapers/CRPITV9Dekker.pdf>

Appendix A: Tables 8 and 9

Table 8: Experimental Results for 10,000 Runs

Poor sensor quality: $q = 0.1$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	1.13	1.143	1.137	1.123	1.226	*1.266	1.261	1.25
0.02	1.071	1.106	1.086	1.047	1.147	*1.213	1.192	1.152
0.05	0.926	1.013	0.964	0.875	0.964	*1.077	1.027	0.938
0.1	0.698	0.857	0.774	0.631	0.684	*0.864	0.778	0.638
0.2	0.388	*0.607	0.488	0.316	0.337	0.541	0.439	0.28
0.5	0.041	*0.147	0.076	0.019	0.022	0.082	0.043	0.011
Moderately poor sensor quality: $q = 0.25$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	1.463	1.461	1.476	1.451	1.866	1.794	*1.907	1.842
0.02	1.388	1.41	1.416	1.374	1.756	1.717	*1.811	1.721
0.05	1.185	1.278	1.245	1.136	1.453	1.512	*1.544	1.386
0.1	0.904	1.097	1.002	0.82	1.052	*1.224	1.179	0.952
0.2	0.514	*0.768	0.633	0.407	0.528	0.758	0.659	0.42
0.5	0.049	*0.19	0.099	0.025	0.031	0.121	0.065	0.015
Moderate sensor quality: $q = 0.5$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	2.283	2.063	2.28	2.219	*3.081	2.459	3.056	2.876
0.02	2.161	1.997	2.186	2.083	2.892	2.355	*2.895	2.674
0.05	1.854	1.83	1.937	1.732	2.406	2.094	*2.485	2.163
0.1	1.402	1.551	1.544	1.251	1.733	1.68	*1.884	1.484
0.2	0.79	*1.091	0.974	0.614	0.857	1.055	1.055	0.652
0.5	0.074	*0.259	0.15	0.037	0.05	0.164	0.099	0.022
Moderately good sensor quality: $q = 0.75$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	3.302	2.579	3.254	3.068	*3.688	2.708	3.597	3.327
0.02	3.14	2.503	3.126	2.883	*3.464	2.6	3.423	3.096
0.05	2.687	2.272	2.758	2.393	2.872	2.294	*2.922	2.487
0.1	2.053	1.94	2.232	1.736	2.085	1.858	*2.236	1.711
0.2	1.13	1.354	*1.389	0.849	1.013	1.153	1.236	0.738
0.5	0.111	*0.33	0.207	0.05	0.062	0.175	0.122	0.03
Good sensor quality: $q = 0.9$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	3.721	2.741	3.635	3.363	*3.766	2.743	3.666	3.378
0.02	3.538	2.654	3.489	3.155	*3.538	2.632	3.484	3.14
0.05	3.024	2.413	*3.076	2.624	2.94	2.324	2.978	2.53
0.1	2.306	2.054	*2.474	1.889	2.117	1.864	2.272	1.727
0.2	1.271	1.427	*1.54	0.931	1.052	1.159	1.249	0.758
0.5	0.118	*0.352	0.228	0.056	0.063	0.172	0.116	0.029

Table 9: Estimated Results Produced by Equation of Best Fit

Poor sensor quality: $q = 0.1$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	1.207	1.376	1.361	1.042	1.41	1.539	*1.541	1.233
0.02	1.121	1.289	1.274	0.956	1.322	1.451	*1.453	1.145
0.05	0.877	1.045	1.03	0.712	1.071	1.199	*1.202	0.894
0.1	0.518	0.686	0.671	0.353	0.701	0.829	*0.831	0.524
0.2	-0.021	0.148	0.133	-0.186	0.14	0.268	*0.270	-0.037
0.5	-0.196	-0.028	-0.043	-0.362	-0.102	0.023	*0.026	-0.279
Moderately poor sensor quality: $q = 0.25$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	1.773	1.919	1.972	1.506	2.084	2.127	*2.217	1.794
0.02	1.682	1.828	1.881	1.415	1.988	2.03	*2.121	1.698
0.05	1.424	1.571	1.624	1.158	1.713	1.756	*1.846	1.424
0.1	1.043	1.19	1.242	0.776	1.304	1.346	*1.436	1.015
0.2	0.461	0.607	0.66	0.194	0.666	0.706	*0.797	0.377
0.5	0.152	0.298	*0.351	-0.115	0.191	0.225	0.317	-0.098
Moderate sensor quality: $q = 0.5$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	2.393	2.409	2.606	2.007	2.818	2.622	*2.888	2.392
0.02	2.295	2.311	2.507	1.909	2.709	2.512	*2.779	2.283
0.05	2.015	2.031	2.228	1.629	2.396	2.198	*2.465	1.97
0.1	1.597	1.613	1.81	1.211	1.922	1.723	*1.99	1.497
0.2	0.941	0.957	1.153	0.554	1.155	0.952	*1.219	0.73
0.5	0.41	0.426	*0.623	0.024	0.292	0.077	0.347	-0.133
Moderately good sensor quality: $q = 0.75$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	2.857	2.697	3.052	2.375	*3.363	2.882	3.338	2.827
0.02	2.752	2.592	2.947	2.27	*3.241	2.759	3.215	2.705
0.05	2.45	2.29	2.645	1.968	*2.889	2.406	2.862	2.353
0.1	1.995	1.835	2.19	1.513	*2.351	1.865	2.322	1.815
0.2	1.264	1.105	*1.46	0.783	1.454	0.962	1.42	0.919
0.5	0.512	0.353	*0.708	0.031	0.204	-0.306	0.156	-0.332
Good sensor quality: $q = 0.9$								
Prob.	Centralised	Distributed	Negotiation	Split	Centr. Share	Distr. Share	Negot. Share	Split Share
0.01	3.094	2.817	3.271	2.562	*3.640	2.976	3.549	3.045
0.02	2.984	2.707	3.161	2.452	*3.510	2.846	3.419	2.915
0.05	2.669	2.392	2.846	2.137	*3.135	2.469	3.042	2.54
0.1	2.192	1.915	2.369	1.659	*2.559	1.888	2.462	1.963
0.2	1.417	1.14	*1.594	0.885	1.584	0.907	1.483	0.989
0.5	0.532	0.255	*0.71	0	0.101	-0.598	-0.017	-0.494

DISTRIBUTION LIST

C4ISR Architectures, Social Network Analysis and the FINC Methodology: An
Experiment in Military Organisational Structure

Anthony H. Dekker

(DSTO-GD-0313)

AUSTRALIA

DEFENCE ORGANISATION

Task Sponsor

COMD DJFHQ

S&T Program

Chief Defence Scientist	} shared copy
FAS Science Policy	
AS Science Corporate Management	
Director General Science Policy Development	
Counsellor Defence Science, London (Doc Data Sheet)	
Counsellor Defence Science, Washington (Doc Data Sheet)	
Scientific Adviser to MRDC Thailand (Doc Data Sheet)	
Scientific Adviser Joint	
Navy Scientific Adviser (Doc Data Sheet and distribution list only)	
Scientific Adviser - Army	
Air Force Scientific Adviser	
Director Trials	

Aeronautical and Maritime Research Laboratory

Director

Electronics and Surveillance Research Laboratory

Director (Doc Data Sheet and distribution list only)

Chief of Information Technology Division

Research Leader, MIE

Research Leader, JS

Research Leader, MSE

Head, Enterprise Visualisation, Instrumentation and
Synchronisation Group (Doc Data Sheet)

Head, Systems Simulation and Assessment Group (Doc Data Sheet)

Head, C3I Operational Analysis Group (Doc Data Sheet)

Head, Information Exploitation Group (Doc Data Sheet)

Head, Intelligence Group (Doc Data Sheet)

Head, Human Systems Integration Group (Doc Data Sheet)

Head, C2 Australian Theatre

Head, Distributed Systems Group (Doc Data Sheet)

Head, C3 Information Systems

Author: Anthony H. Dekker (20 copies)

Publications & Publicity Officer, ITD / Executive Officer, ITD 1 shared copy

DSTO Library and Archives

Library Fishermans Bend (Doc Data Sheet)

Library Maribyrnong (Doc Data Sheet)

Library Edinburgh

Australian Archives

US Defense Technical Information Center, 2 copies

UK Defence Research Information Centre, 2 copies

Canada Defence Scientific Information Service, 1 copy

NZ Defence Information Centre, 1 copy

National Library of Australia, 1 copy

Capability Systems Staff

Director General Maritime Development (Doc Data Sheet only)

Director General Land Development

Director General Aerospace Development (Doc Data Sheet only)

Army

ABCA Standardisation Officer, Land Warfare Development Sector Tobruk Barracks, Puckapunyal, 3662 (4 copies)

SO (Science), Deployable Joint Force Headquarters (DJFHQ) (L), MILPO Gallipoli Barracks, Enoggera QLD 4052

Director, Force Development Group, Combined Arms Training and Development Centre, MILPO Tobruk Barracks, Puckapunyal, 3662

Intelligence Program

DGSTA Defence Intelligence Organisation

Manager, Information Centre, Defence Intelligence Organisation

Corporate Support Program

Library Manager, DLS-Canberra

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy

Library

Head of Aerospace and Mechanical Engineering

Hargrave Library, Monash University (Doc Data Sheet only)

Librarian, Flinders University

OTHER ORGANISATIONS

NASA (Canberra)

AusInfo

State Library of South Australia

OUTSIDE AUSTRALIA

ABSTRACTING AND INFORMATION ORGANISATIONS

Library, Chemical Abstracts Reference Service
Engineering Societies Library, US
Materials Information, Cambridge Scientific Abstracts, US
Documents Librarian, The Center for Research Libraries, US

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, US

SPARES (5 copies)

Total number of copies: 70

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA									
				1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)					
2. TITLE C4ISR Architectures, Social Network Analysis and the FINC Methodology: An Experiment in Military Organisational Structure			3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION) Document (U) Title (U) Abstract (U)						
4. AUTHOR(S) Anthony H. Dekker			5. CORPORATE AUTHOR Electronics and Surveillance Research Laboratory PO Box 1500 Edinburgh South Australia 5111 Australia						
6a. DSTO NUMBER DSTO-GD-0313		6b. AR NUMBER AR-012-129		6c. TYPE OF REPORT General Document		7. DOCUMENT DATE January 2002			
8. FILE NUMBER N9505/23/19		9. TASK NUMBER ARM 01/292		10. TASK SPONSOR COMD DJFHQ		11. NO. OF PAGES 39		12. NO. OF REFERENCES 18	
13. URL on the World Wide http://www.dsto.defence.gov.au/corporate/reports/DSTO-GD-0313.pdf						14. RELEASE AUTHORITY Chief, Information Technology Division			
15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT <i>Approved for public release</i>									
OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111									
16. DELIBERATE ANNOUNCEMENT No Limitations									
17. CASUAL ANNOUNCEMENT Yes									
18. DEFTTEST DESCRIPTORS Command, control, communications, computers and intelligence, surveillance and reconnaissance Network analysis Information flow Technological intelligence Social networks									
19. ABSTRACT In this paper we present a simple testbed for experimenting with C4ISR architectures (based on a "SCUD hunt" scenario), the FINC methodology for analysing C4ISR architectures, and some experimental results. The testbed allows us to explore different organisational architectures under a range of conditions. The FINC (Force, Intelligence, Networking and C2) methodology allows the calculation of three metrics for every C4ISR architecture. Applying the FINC methodology to our testbed provides a partial validation of the methodology, as well as allowing us to derive four basic principles of C4ISR architectures.									